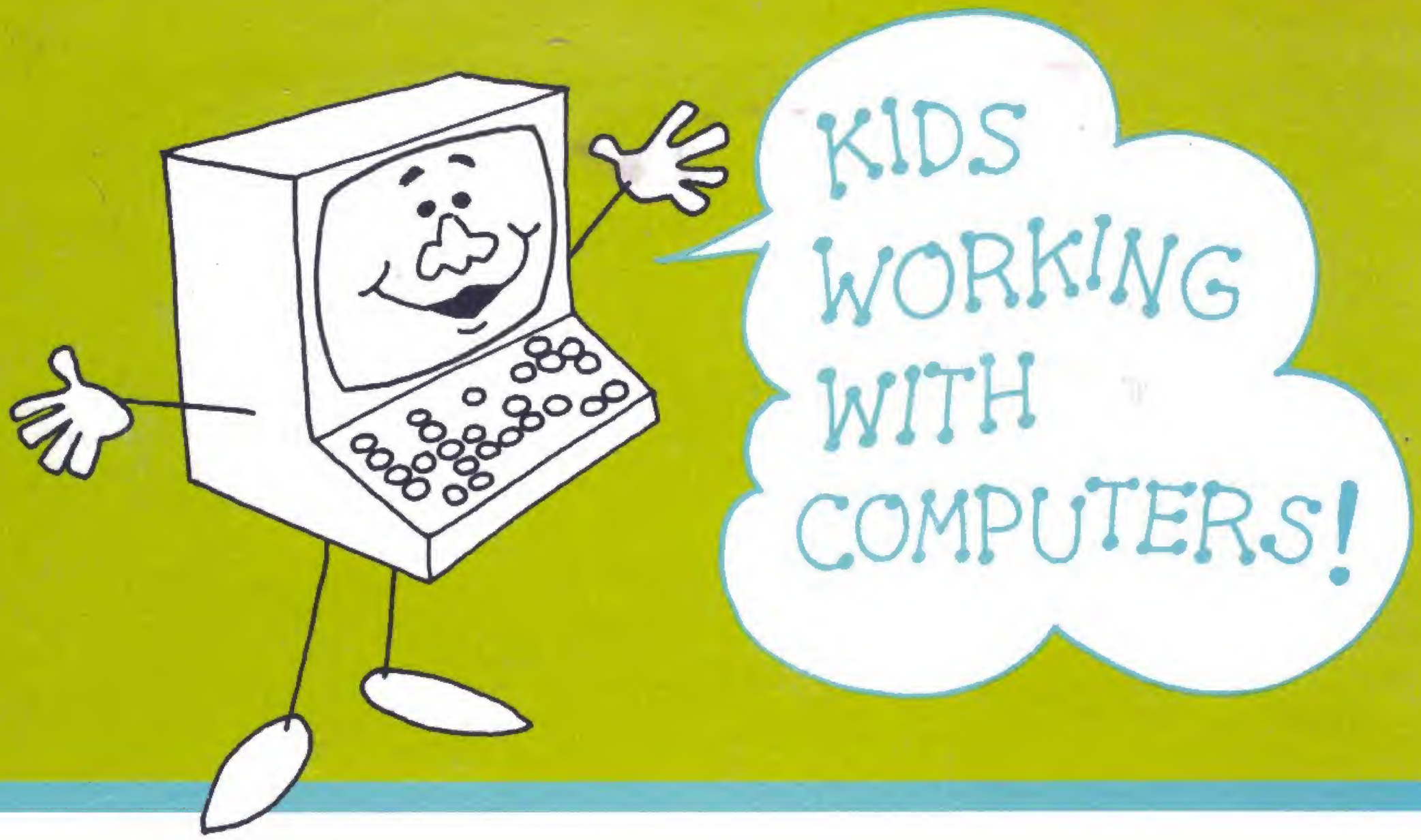


THE TRS-80® BASIC MANUAL

Thomas Milton Kemnitz
Lynne Mass

Kids Working with Computers

THE TRS-80® BASIC MANUAL

Thomas Milton Kemnitz **Lynne Mass**

CHILDRENS PRESS™
CHICAGO

Library of Congress Cataloging in Publication Data

Kemnitz, Thomas Milton.
The TRS-80 BASIC manual.

(Kids working with computers)
Includes index.
Summary: Explains how to use the computer language BASIC to write fundamental programs for the TRS-80 computer.
1. TRS-80 computers—Programming—Juvenile literature. 2. BASIC (Computer program language)—Juvenile literature. [1. TRS-80 computers—Programming. 2. BASIC (Computer program language) 3. Programming (Computers) 4. Computers]
1. Mass, Lynne. II. Title. III. Series.

QA76.8.T18K46 1985 001.64'2 85-3735
ISBN 0-516-08431-3

Library bound edition published by Childrens Press under license from Trillium Press.

1 2 3 4 5 6 7 8 9 10 R 94 93 92 91 90 89 88 87 86 85

CONTENTS

1

Meet the Computer

The best way to get to know your computer is to start working with it:

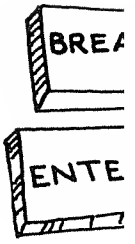
1. First hold down the **BREAK** button, which is at the top right of the keyboard. Then turn the ON switch at the bottom right of the computer.
2. When you see the blinking light **(cursor),** push the **ENTER** button.
3. The screen will say,

   ```
   ''Memory Size?''
   ```

   Push ENTER again.
4. This time the screen will show you:

   ```
   READY
   >
   ```

   □ — (blinking cursor)
5. Type **CLS** (clear screen) and push ENTER.



You are now ready for your first **program.**

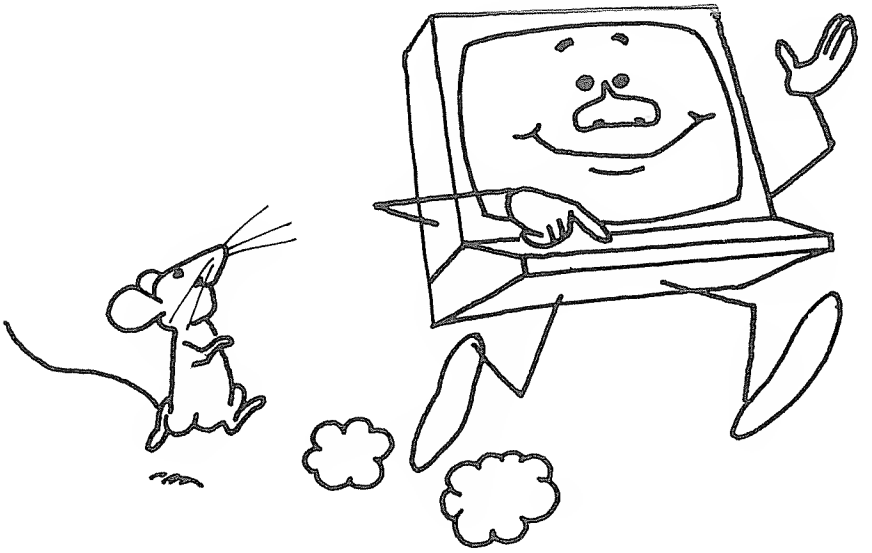
Type exactly what you see. If you type a mistake, push ← and type again. Remember to push ENTER at the end of each line.

```
10 PRINT ''HELLO, MY NAME IS (type in name).''
20 PRINT ''I AM LEARNING TO WORK THE TRS-80.''
30 PRINT ''MY TEACHER IS (type in name).''
40 PRINT ''(make up something about yourself).''
```

Now that you have typed your program, type **LIST** and push ENTER. What happens?

This time try typing **RUN**. Did you remember to push ENTER?

What happens if you type **NEW** [ENTER] and then **RUN** [ENTER]?



Review:

What do these words do:

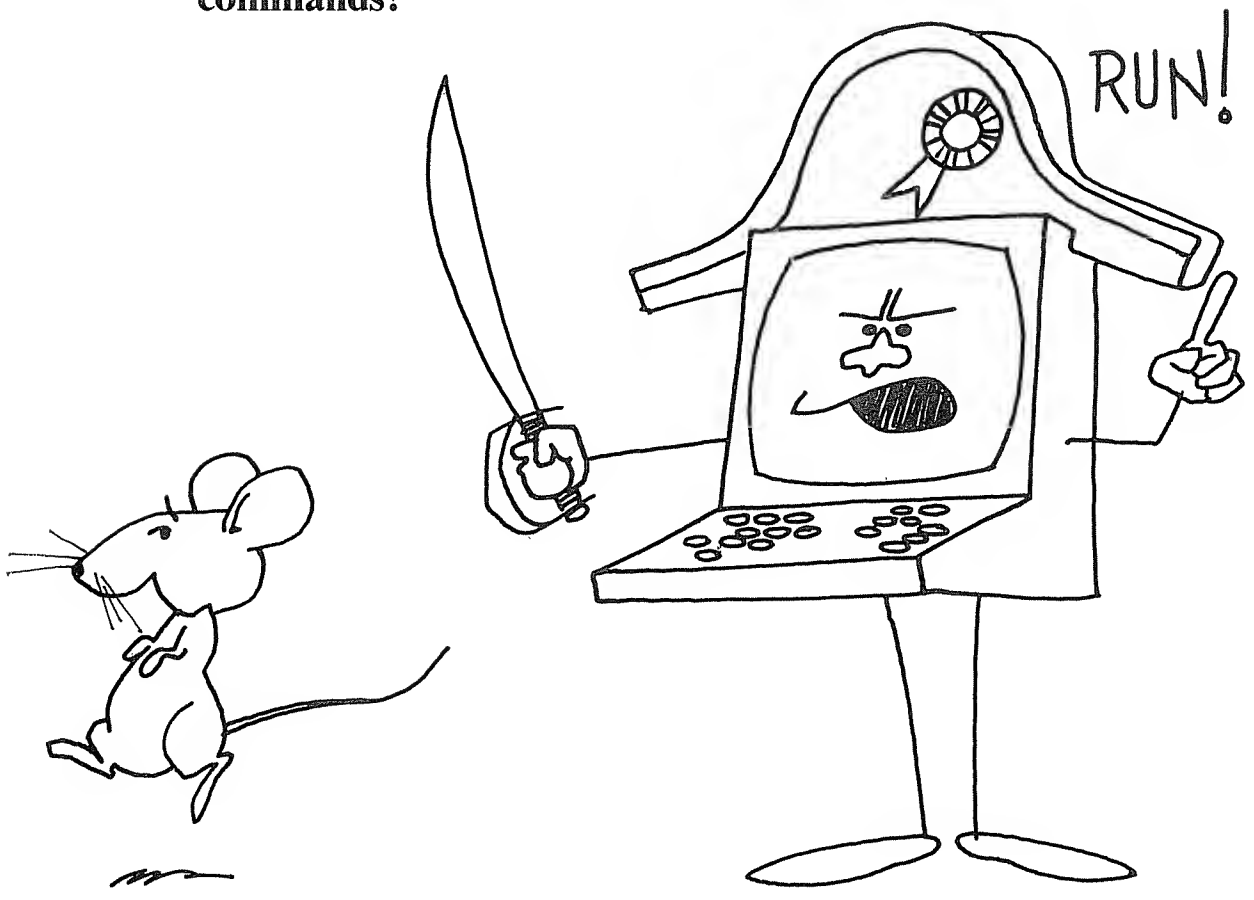
CLS

LIST

RUN

NEW

Why are these words called **commands?**



2

Do I Always Have to Type PRINT?

Type the following programs. Always remember to push ENTER.

```
10 PRINT ''I AM TYPING...''
20 PRINT ''ON A COMPUTER.''
30 PRINT ''THIS IS FUN!''
```

Type LIST. Now type RUN. What is the difference between LIST and RUN?

Type in the same program again, but on line 20 type ? instead of the word PRINT. Now type LIST. What happened when you did this?

Try your own ideas and decide whether you want to use ? or PRINT.

3

Do I Need Quotation Marks?

Type the following program:

```
10 PRINT ''HI MOM!''
20 PRINT ''HI DAD!''
```

Don't forget LIST and RUN.

Now try the same program again, this time without quotation marks.

```
10 ? HI MOM!
20 ? HI, DAD!
```

How are these two programs different?

Just for Fun

```
FOR B = 1 TO 832 : ? ''A'' ; : NEXT B : ?''PHEW!''
```

4

Computer Loops

Type the following program and RUN it:

```
10 PRINT ''HELP—MY COMPUTER WENT CRAZY!''
20 GOTO 10
```

When you have seen enough, push **BREAK**. This is called a **loop** and is controlled by the **GOTO** command.

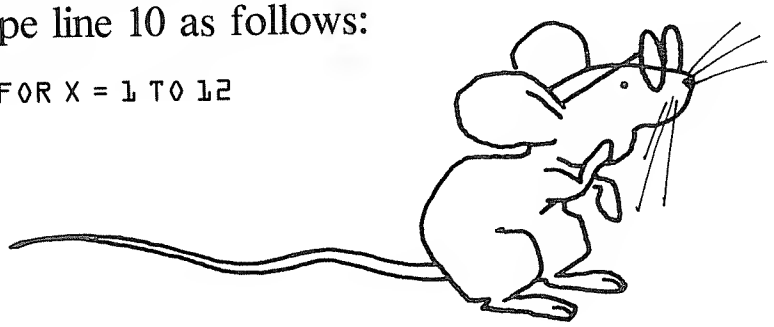
Now try this program:

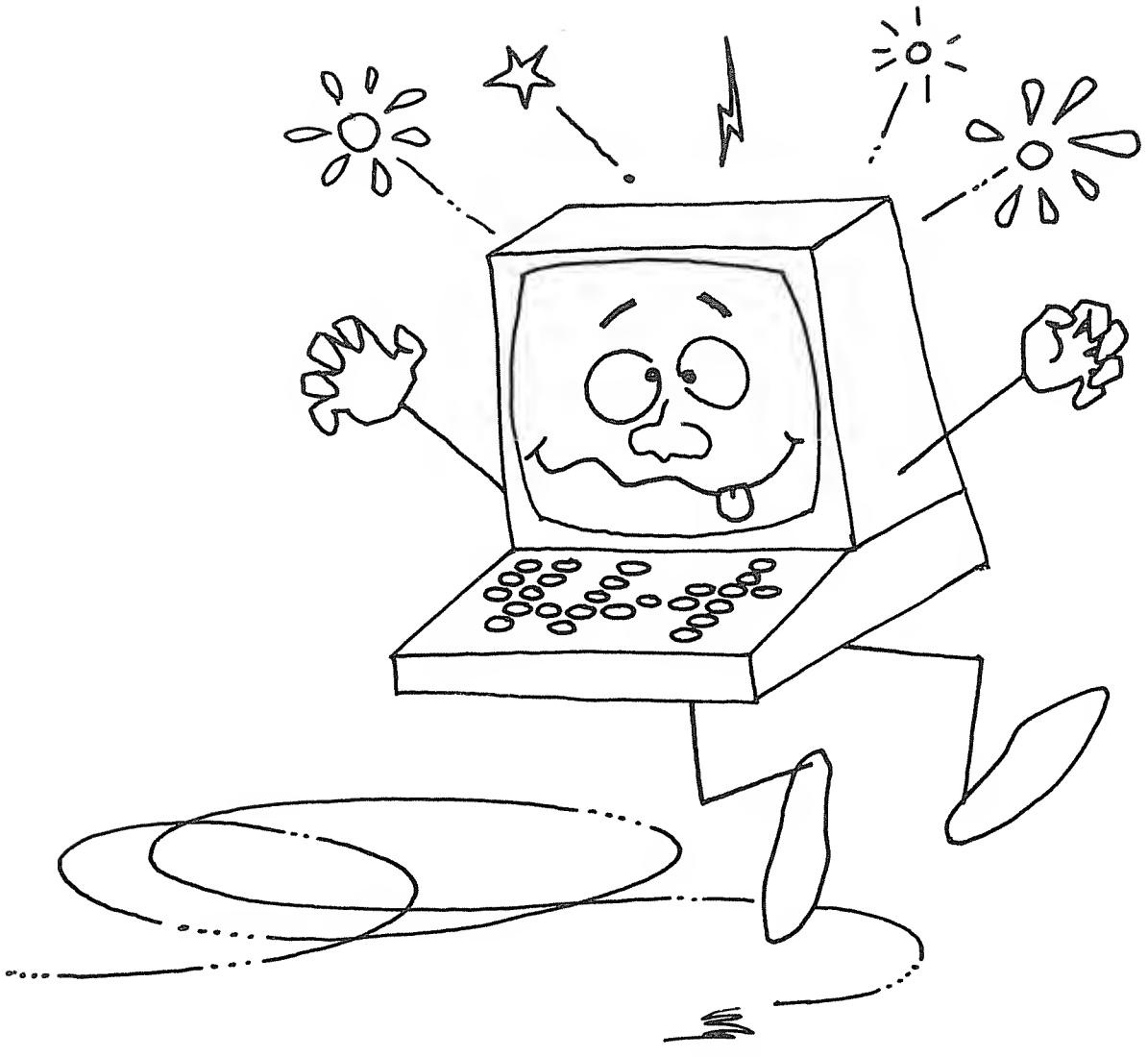
```
10 FOR X = 1 TO 5
20 PRINT ''HELP—MY COMPUTER WENT CRAZY.''
30 NEXT X
40 PRINT ''NO, IT IS UNDER CONTROL.''
```

Lines 10 through 30 of this program are called a **FOR/NEXT** loop. The statement in line 10 says, "I am going to do something five times." The statement in line 20 tells what will be done. The statement in line 30 sends the computer back to the counter.

Retype line 10 as follows:

```
10 FOR X = 1 TO 12
```





This will erase what was there before, just as on a tape recorder.

What do you think will happen after you type RUN? Try it. What did happen?

Put your name in line 20. Have fun!

5

The Egg Timer

Type the following program:

```
10 PRINT ''WAIT RIGHT HERE!''
20 FOR X = 1 TO 3000
30 NEXT X
40 PRINT ''YOUR TIME IS UP.''
```

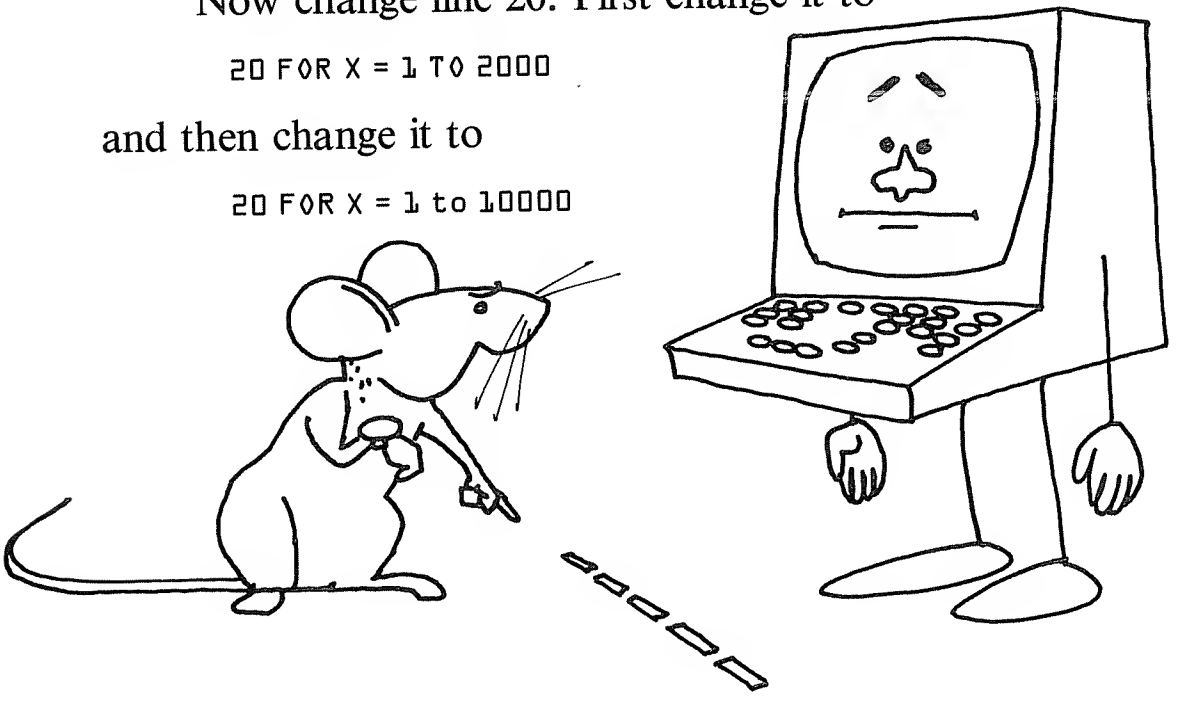
Did time go by between the printing of "WAIT RIGHT HERE!" and "YOUR TIME IS UP."? How much time went by?

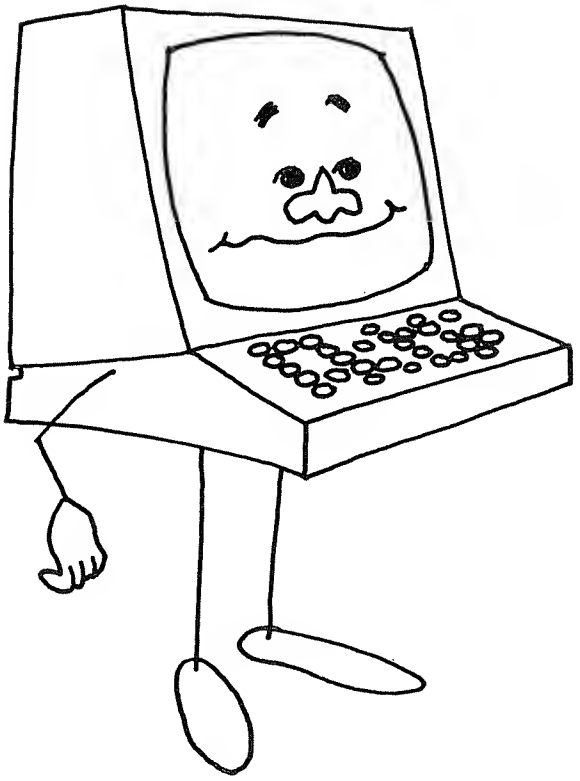
Now change line 20. First change it to

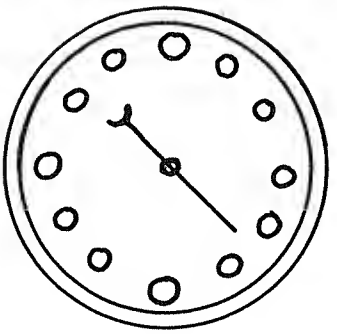
```
20 FOR X = 1 TO 2000
```

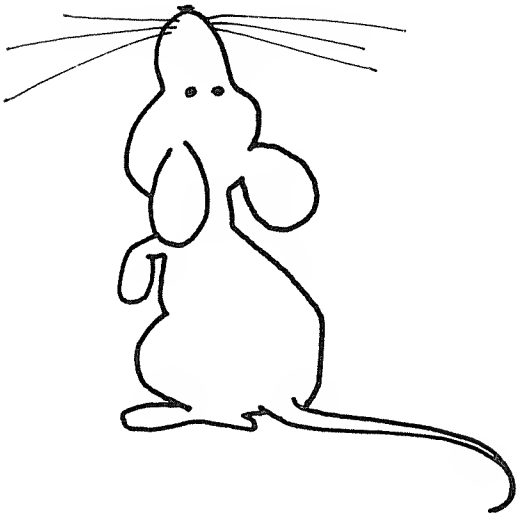
and then change it to

```
20 FOR X = 1 to 10000
```









How are the two times different? Is one longer than the other? If so, which one?

Line 20 is the **timer.** It tells the computer how much time to leave.

Can you change the program to make a thirty-second delay? How? Try it.

6

How to Boss Around Your Computer

Type the following program:

```
10 FOR X = 1 TO 50
20 PRINT ''NOW I AM PROGRAMMING.''
30 PRINT ''I AM STILL AT IT.''
40 PRINT ''AND I CAN'T STOP.''
50 NEXT X
```

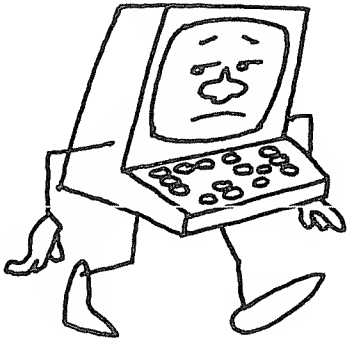
What happened? How many times do you think you told the computer to write lines 20, 30, and 40?

Keep the same program and add this line:

```
45 PRINT: PRINT
```

Does the program look different? How?

Did you have to retype the whole program to add a line? Explain.



7

Line Order

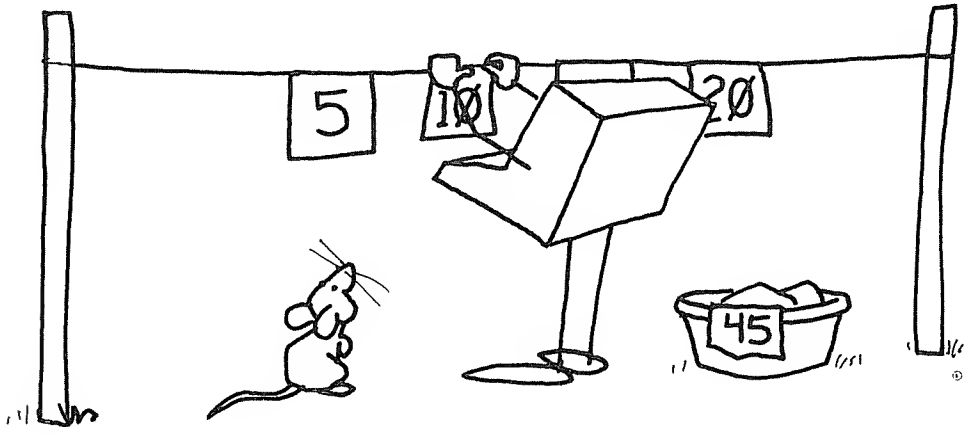
Type the following program:

```
10 ?''I AM USING THE COMPUTER.''
20 ?''AND I DON'T WANT TO STOP.''
 5 ?''NOW I AM PROGRAMMING.''
```

Now LIST the program. What did the computer do to help you?

Can you figure out a way to add more lines between the beginning and the end? What is the way you decided to use?

What do you think would happen if you were to give two lines the same number? Try it.



8

Computer Acrobatics

Type the following program.

NOTE: This program does not work on the color computer.

```
10 CLS
20 ?@ 204, ''(write your first name)''
30 ?@ 407, ''(write your middle name)''
40 ?@ 609, ''(write your last name)''
```

When you RUN the program, where does your name appear on the screen?

For Model 100, use this program:

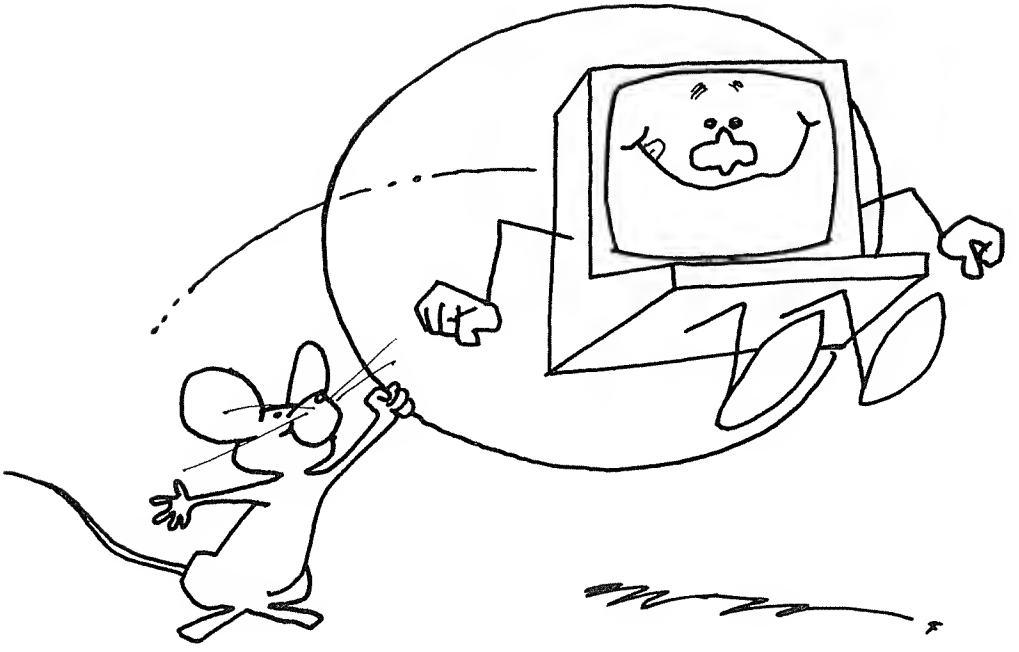
```
10 ?@ 005, ''(write your first name)''
20 ?@ 088, ''(write your middle name)''
30 ?@ 211, ''(write your last name)''
```

Do you remember how to make your name repeat itself? Try it.

Did you type:

```
50 GOTO 20
```

Stop this by pushing BREAK. To continue, type **CONT.**



Try stopping the program several times. Can you get it to break at different points?

Try this program. Type NEW and then type:

```
10 CLS
20 ?''(YOUR NAME)'';
30 GOTO 20
```

RUN it. Use BREAK to stop the program when you've seen enough.

Now try:

```
20 ?''(YOUR NAME)'',
```

What is the difference between a semicolon (;) and a comma (,) after a statement that you tell the computer to print?

9

Let's Get Fancy

Program the computer with:

```
10 PRINT ''RUBBER BABY BUGGY BUMPERS.''
```

Clear the screen by typing **CLS.**

Does this erase your program? How can you test it? Yes, you can see what is in the memory by typing LIST.

Now type in this command:

```
NEW
```

Does this also clear the screen? Is your program erased?

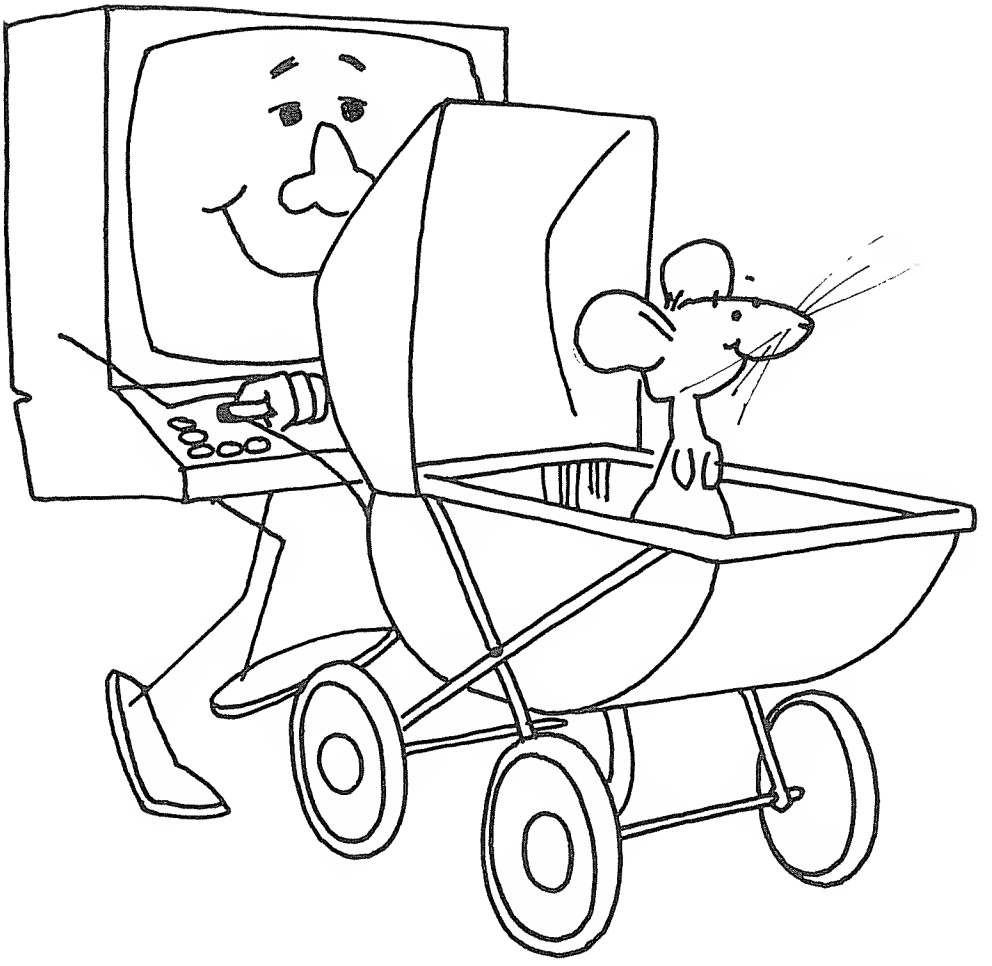
Type the program again. Hold down the left SHIFT and press → What happens?

NOTE: This does not work on the color computer or Model 100.

Try typing LIST. What happens?

Now try RUN. What happens?

Try pressing CLEAR. What happens?



Try RUN

[SHIFT]→

RUN

What is happening?

10

String a Design

A **string**—a **variable** and a $—is one or more characters. A string can be interesting. For example, try typing in this:

```
02 CLS
05 A$ = ''GOOD MORNING''
10 PRINT A$
```

What happens?

You can even get a number value for a string. LEN is a code that will tell you how many characters—in this case, letters and spaces—are in the string.

Try this:

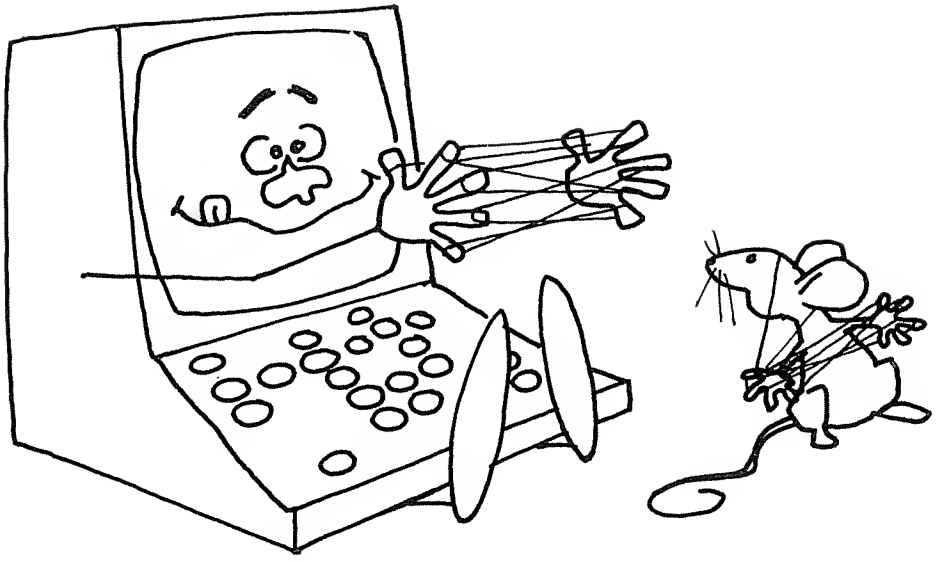
```
10 PRINT LEN (A$), LEN (''YES'')
```

What do the 12 and 3 represent? *(CLUE: LEN is a computer word for "length.")*

Now try this:

```
10 PRINT LEFT$ (A$,4)
```

What do you get? It's good if you see GOOD. What do you think LEFT does?



Now try this:

```
10 FOR N = 1 TO LEN (A$)
20 PRINT LEFT$ (A$,N)
30 NEXT N
```

Isn't that cute?

Why does this happen? Since A$ has twelve characters, this loop will be done twelve times, with N = 1,2,3,. . .11,12. The first time, only the first character will be printed (and N = 1). The second time the first two characters will be printed (and N = 2), and so on.

Now try the same thing with a right $, using the example above.

11

Pick and Choose

Here is a program to try on a friend. Try doing it yourself first:

```
 2 CLS
 5 PRINT ''TYPE IN YOUR AGE.''
10 INPUT A
15 PRINT ''YOUR AGE IS''
20 PRINT A
RUN
```

INPUT asks your friend to type in a number while the program is running. When you are finished with this program, type NEW.

Now try this easy guessing game:

```
 2 CLS
 5 PRINT ''PICK A NUMBER. TYPE IT IN. THE CHOICES
   ARE 1, 2, OR 3.''
10 INPUT N
20 IF N = 3 THEN PRINT ''YOU ARE CORRECT.'' : END
30 IF N < 3 THEN PRINT ''NO, TRY AGAIN.''
```

What happens if you add:

```
35 GOTO 10
```

12

Getting Flashy

Using the program on the preceding page, what happens if you type:

```
20 IF N < > 3 THEN PRINT ''NO, TRY AGAIN.''
25 GOTO 5
30 IF N = 3 THEN PRINT ''YOU ARE CORRECT.''
35 END
```

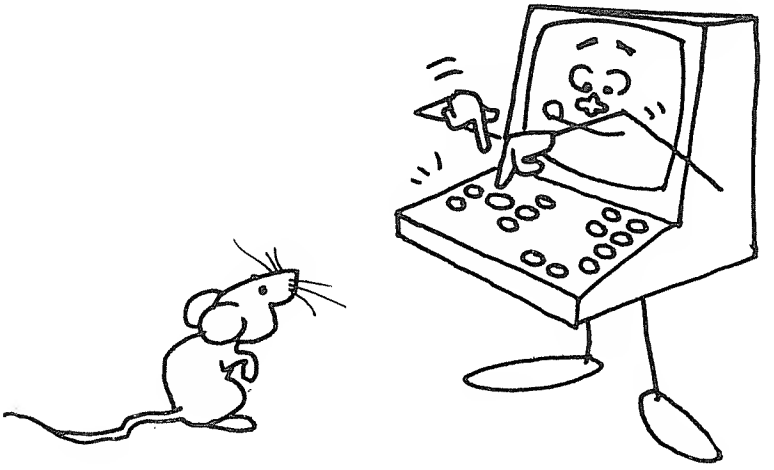
What happens if you take out line number 25 by typing in:

```
25 [ENTER]
```

and instead type:

```
20 IF N <> 3 THEN PRINT ''NO, TRY AGAIN.'': GOTO 5
```

Why did you have to use BREAK to make these changes?



13

Be Creative

At this point, you have learned enough to go off on your own. Experiment.

Develop your own programs.

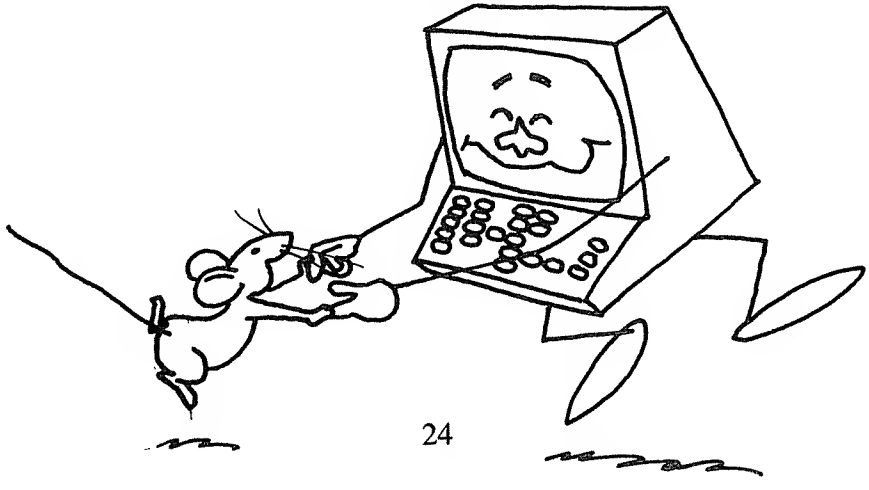
Try out your ideas.

Write down your ideas as you think of them.

Share your programs with friends. Try running one another's programs. Talk about what happens.

Work together to develop new programs.

Have fun with your computer.



14

The Ticking TRS-80®

Would you like to know how long it has been since you turned the computer on?

```
PRINT TIME$
```

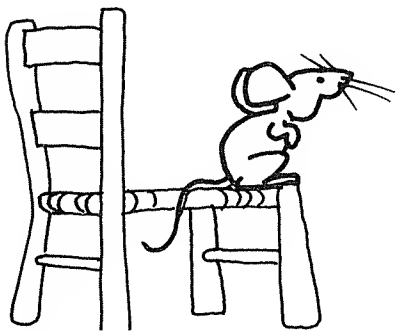
If you want the computer to tell you the time, type in the following program:

NOTE: This program will not work on the color computer and Model 100.

```
 2 CLS
10 DEFINT A-Z
20 DIM TM(5)
30 CL = 16924
40 PRINT ''ENTER DATE & TIME AS 24 HOUR CLOCK: MO,
   DA, YR, HR, MN, SS''
50 INPUT TM(0), TM(1), TM(2), TM(3), TM(4), TM(5)
60 FOR I = 0 TO 5
70 POKE CL-I, TM(I)
80 NEXT I
90 ? ''CLOCK IS RUNNING''
100 END
```

Now type:

```
PRINT TIME$
```



15

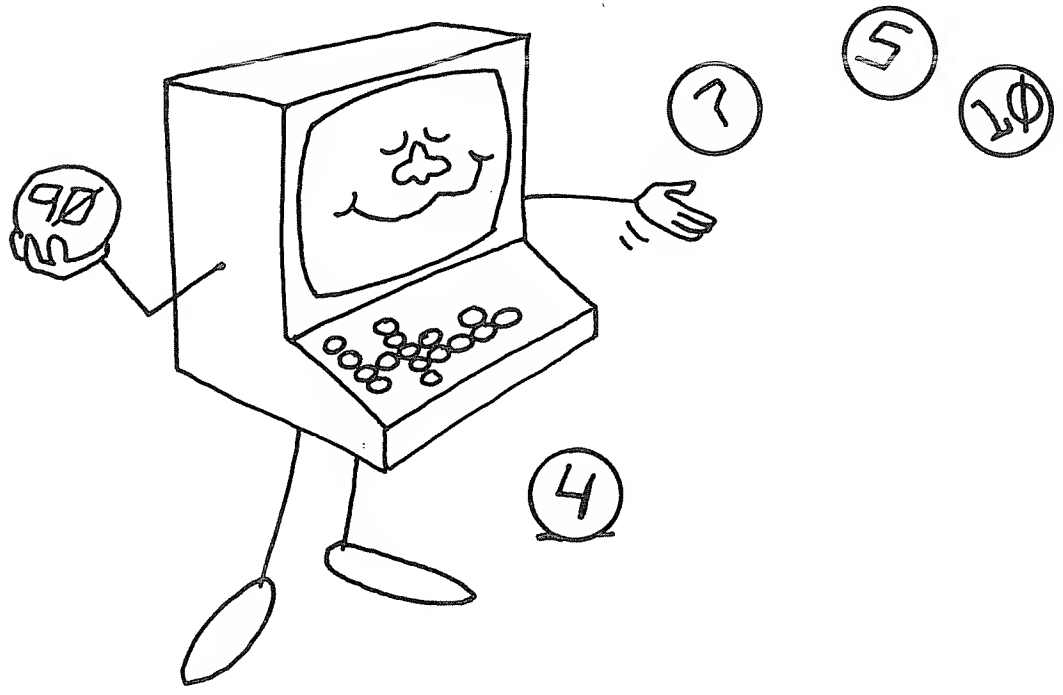
At Random

The computer can print out numbers that are not predictable. These form the basis for many computer games. The numbers are formed randomly and have a code—**RND**, which stands for "**random numbers.**"

To see what RND(x) does, try this:

```
PRINT RND (0)
```

NOTE: On the Model 100, you must use RND(1) whenever the program calls for RND (0).



What did you get? Try it again. Did you get the same number? Are the numbers whole numbers or decimals?

How can you get a number > 1 (greater than 1)? Try this:

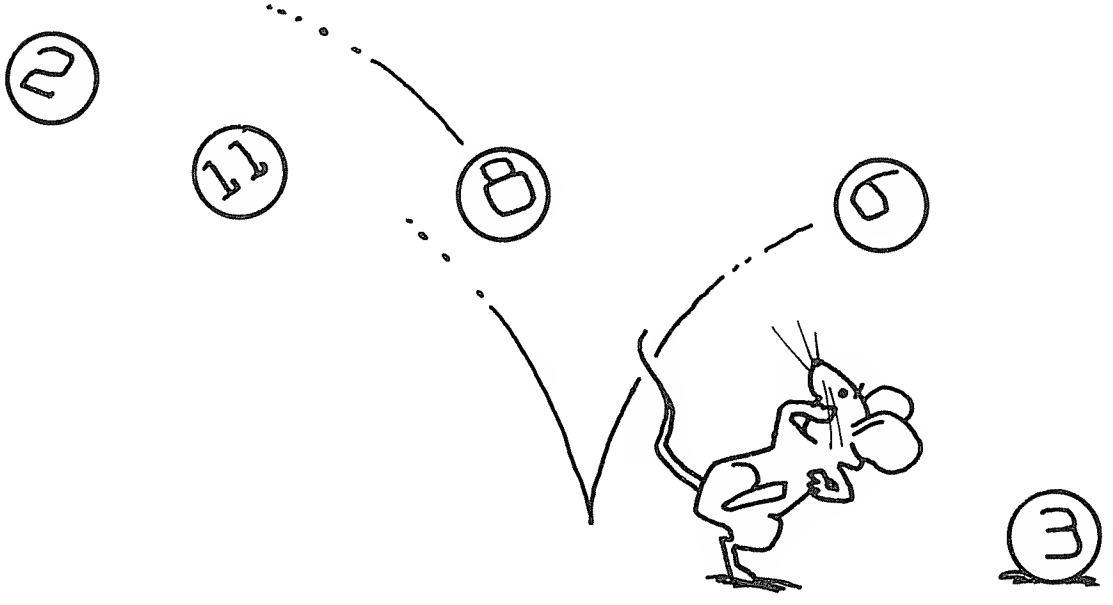
```
PRINT 10 * RND (0)
```

To get rid of the numbers after the decimal, try this:

```
PRINT INT (10 * RND (0) )
```

INT is a code for "**integer**".

What happens if you multiply by 100? Experiment. Change RND(0) to RND (other things).



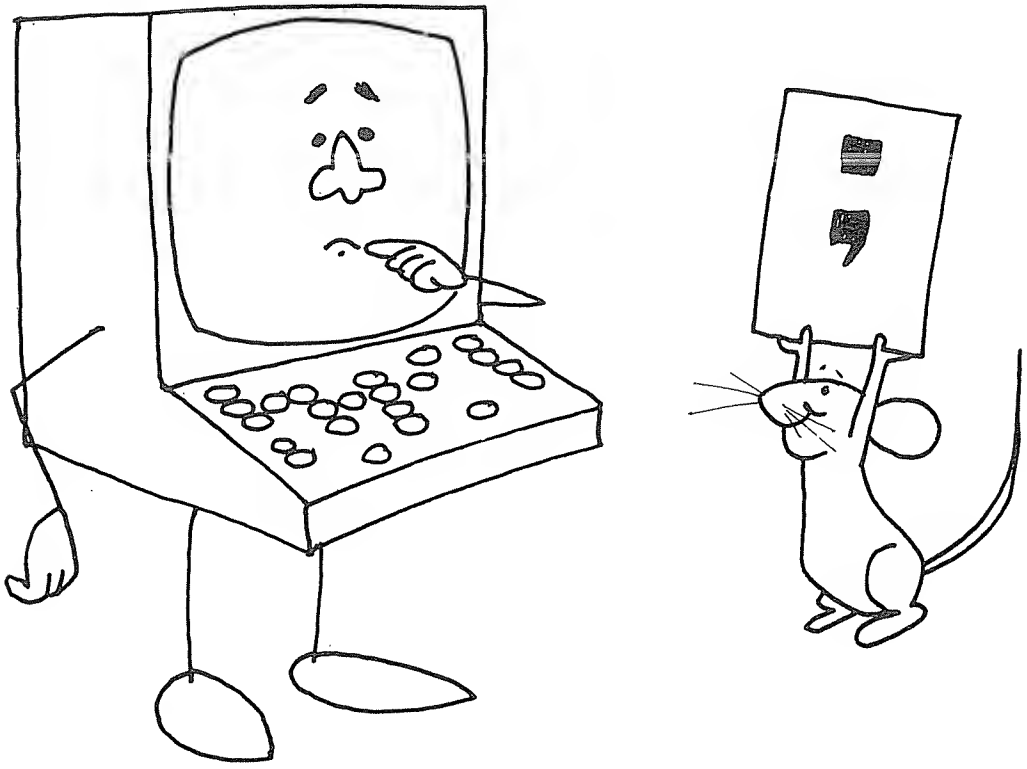
Try this program. Before you run it, predict what will happen.

```
 2 CLS
10 FOR N = 1 TO 10
20 PRINT RND (0)
30 NEXT N
```

Change lines 10 and 20 to:

```
10 FOR N = 1 TO 90
20 PRINT RND (0);
```

What happens? What does the semicolon (;) do?



Here is a number game using RND and other things you have learned:

```
 02 CLS
 10 G = 1 + INT (100 * RND (0) )
 15 PRINT ''GUESS A NUMBER FROM 1 TO 100.''
 18 C = 0
 20 INPUT R
 50 C = C + 1
 60 IF R = G THEN 130
 70 IF R < G THEN 100
 80 PRINT ''TOO HIGH''
 90 GOTO 110
100 PRINT ''TOO LOW''
110 GOTO 20
130 PRINT ''VERY GOOD, YOU GUESSED IT.''
140 PRINT ''YOU TOOK 'C' GUESSES.''
```

Make up your own guessing game. SAVE it on a disk or on the printer.

16

Repeat

One advantage of computers is their ability to perform repetitive tasks quickly and without getting bored.

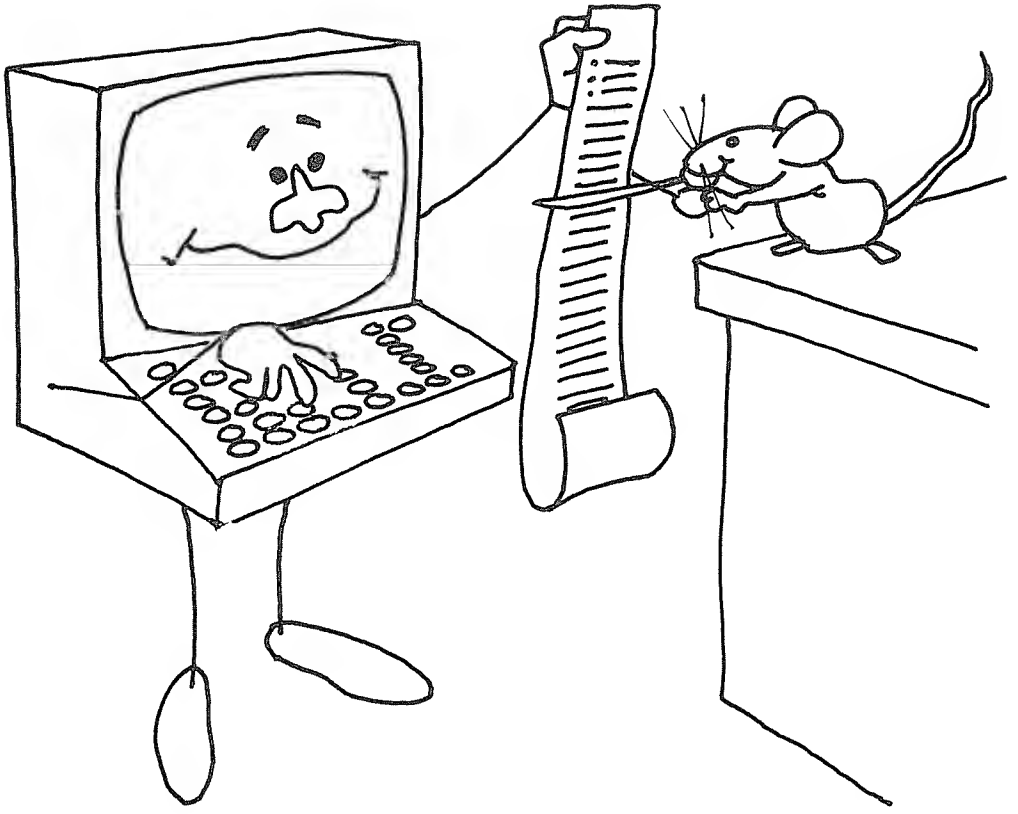
Follow these programs involving square roots and see the shortcuts that develop. SQR is computer language for "square root" and SQR (x) is the way you write "square root of a number," with x standing for the number.

Try this program:

```
10 PRINT 1, SQR (1)
20 PRINT 2, SQR (2)
30 PRINT 3, SQR (3)
40 PRINT 4, SQR (4)
50 PRINT 5, SQR (5)
60 PRINT 6, SQR (6)
70 PRINT 7, SQR (7)
80 PRINT 8, SQR (8)
90 PRINT 9, SQR (9)
100 PRINT 10 SQR (10)
```

This a shortened form of the same program. Try it. Type NEW and then type:

```
10 N = 1
20 PRINT N, SQR (N)
30 N = N + 1
40 IF N < = 10 THEN GOTO 20
```



How does this four-line program compare with the ten-line program you ran first?

Using the loop in the last program, we can shorten the program even more, using a FOR/NEXT statement. Type NEW and then type:

```
10 FOR N = 1 TO 10
20 PRINT N, SQR (N)
30 NEXT N
```

Can you figure out a program for printing a table of square roots for only the even integers from 10 to 20?

17

Make a Guessing Game

The purpose of this program is to play a little game in which you try to guess one of the numbers in lines 110 and 120. These are called **DATA** statements.

As you type in the program, try to decide what you think is the purpose of **READ**:

```
  2 CLS
 10 PRINT ''PICK A NUMBER.''
 20 INPUT G
 30 READ D
 40 IF D = -99999 THEN GOTO 80
 50 IF D < > G THEN GOTO 30
 60 PRINT ''YOU ARE CORRECT.''
 70 END
 80 PRINT ''WRONG, TRY AGAIN.''
 90 RESTORE
100 GOTO 10
110 DATA 9, 15, 18, -60 242, 0, 80
120 DATA 78, 4, 15, 25, -22, -99999
```

DATA statements may be placed anywhere in the program. DATA is information read by the computer when a READ statement appears.

Now make up your own guessing game.

18

Math Magic

Our computer can do many tricks with numbers. Let's get used to some easy ones first.

Try this:

```
PRINT 3 + 4
```

The TRS-80® can do six different arithmetic operations. Try them all:

1. addition (+)
2. subtraction (–)
3. multiplication (*)
4. division (/)
5. exponentiation (4^5, for example)
6. square root ($\sqrt{16}$, for example)

```
PRINT 5 + 7
PRINT 6 - 2
PRINT 7 * 8
PRINT 63/7
PRINT 4*4*4*4*4
  or
PRINT 4↑5
PRINT SQR (16)
```

Make up some math problems of your own.

19

Some Strange Numbers

Type:

```
PRINT 4.340
```

What do you see? The computer does not print the end zero—or a beginning one either.

To get an idea of what computers do with numbers in expressions, do these in your head or on paper first, and then try them on the computer:

1. 3+2
2. 4+6−2+1
3. 8*4
4. 4↑2+1
5. 6/4+1
6. 5−4/2
7. 4/2−2
8. 6*6*7+1
9. 2↑2↑3+1
10. 2*2*3+1
11. 2*2+*3
12. 2*2*1+3
13. 8/2/2/1
14. 8*2/2+3
15. 20/2*5

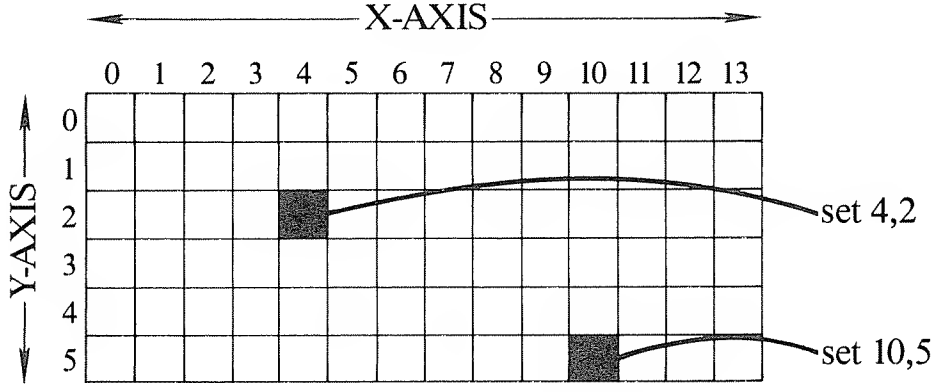
How could you write these expressions more clearly, using parentheses?

20

Great Graphics

To get graphics on the computer, you have to draw pictures with dots and lines.

Part of the screen looks like this:



The TRS-80® has 48 sectors on the Y-axis (vertical) and 128 on the X-axis (horizontal).

To find a particular spot on the screen, try this program:

```
 2 CLS
10 LET X = 64
20 LET Y = 24
30 SET (X,Y)
```

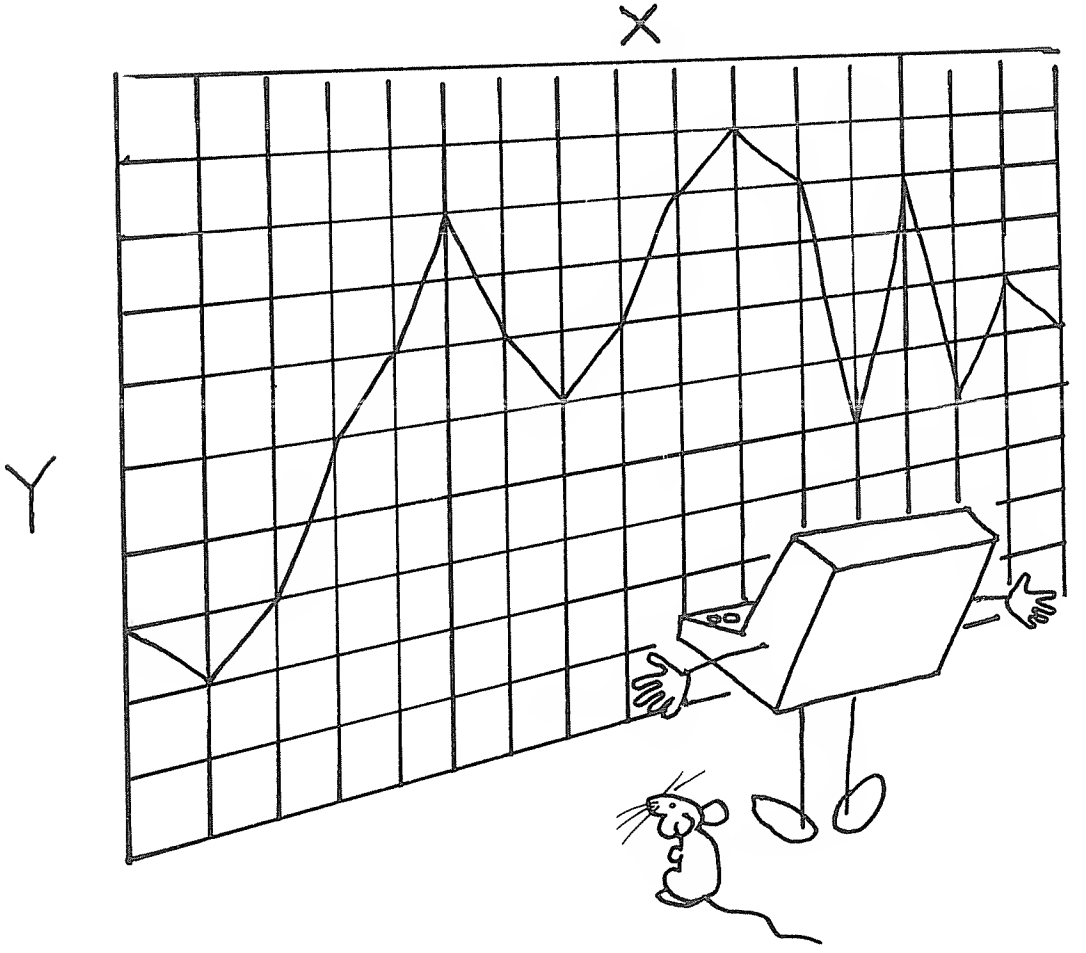
Now try adding these lines:

```
40 RESET (X,Y)
50 GOTO 2
```

To stop the program, push the BREAK key.

What happens if you add:

```
25 CLS
50 GOTO 25
```



21

Going Straight

Suppose you want a straight line. There has to be an easier way than lighting up little squares one after another. It is done in the same way you learned to get a list of numbers in lesson 16. All you have to do is give the command for the X- or Y-axis in the form:

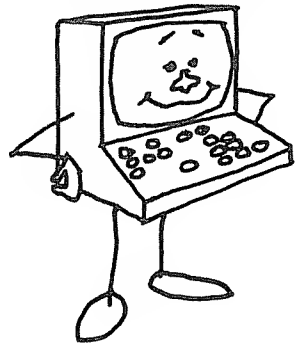
```
FOR X = 0 TO 127
```

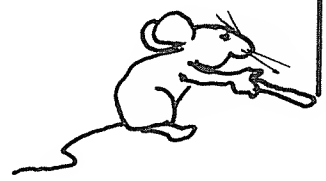
Now try this program:

```
 2 CLS
10 FOR X = 0 TO 127
20 FOR Y = 0 TO 1
30 SET (X,Y)
40 NEXT X
50 GOTO 10
```

To move the line, just change the numbers.

How would you draw a vertical line? Try it.



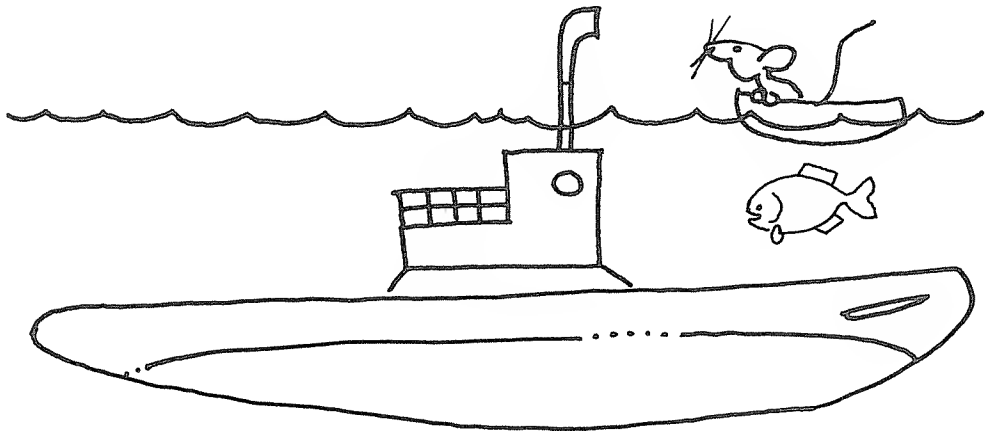


22

GOSUB

Now that you have learned to draw one line, the question is: How can we draw a lot of lines to make a picture? You do this by using the **GOSUB** and **RETURN** commands. GOSUB is a way of going to a routine that will be used many times in a program. It is like a string variable, but longer. Try it:

```
  2 CLS
 10 PRINT ''WE WERE DOING GRAPHICS.''
 20 GOSUB 100
 30 PRINT ''NOW WE'LL DO GRAPHICS AGAIN.''
 40 END
100 PRINT ''BUT WE NEED TO KNOW A NEW PROGRAMMING
    TRICK.''
110 PRINT ''SO WE ARE GOING SUB.''
120 RETURN
```



23

Getting Framed

Now let's try drawing more than one line. In fact, let's make a frame. Here's how:

```
  2 CLS
 10 GOSUB 100
 20 FOR X = 0 TO 127
 30 FOR Y = 47 TO 47
 40 SET (X,Y)
 50 NEXT X
 60 GOSUB 200
100 FOR X = 0 TO 127
110 FOR Y = 0 TO 1
120 SET (X,Y)
130 NEXT X
140 RETURN
200 FOR X = 0 TO 1
210 FOR Y = 0 TO 47
220 SET (X,Y)
230 NEXT Y
240 GOSUB 300
300 FOR X = 127 TO 127
310 FOR Y = 0 TO 47
320 SET (X,Y)
330 NEXT Y
340 GOTO 10
```

What happens if you change line 140 to:

```
GOTO 2
```

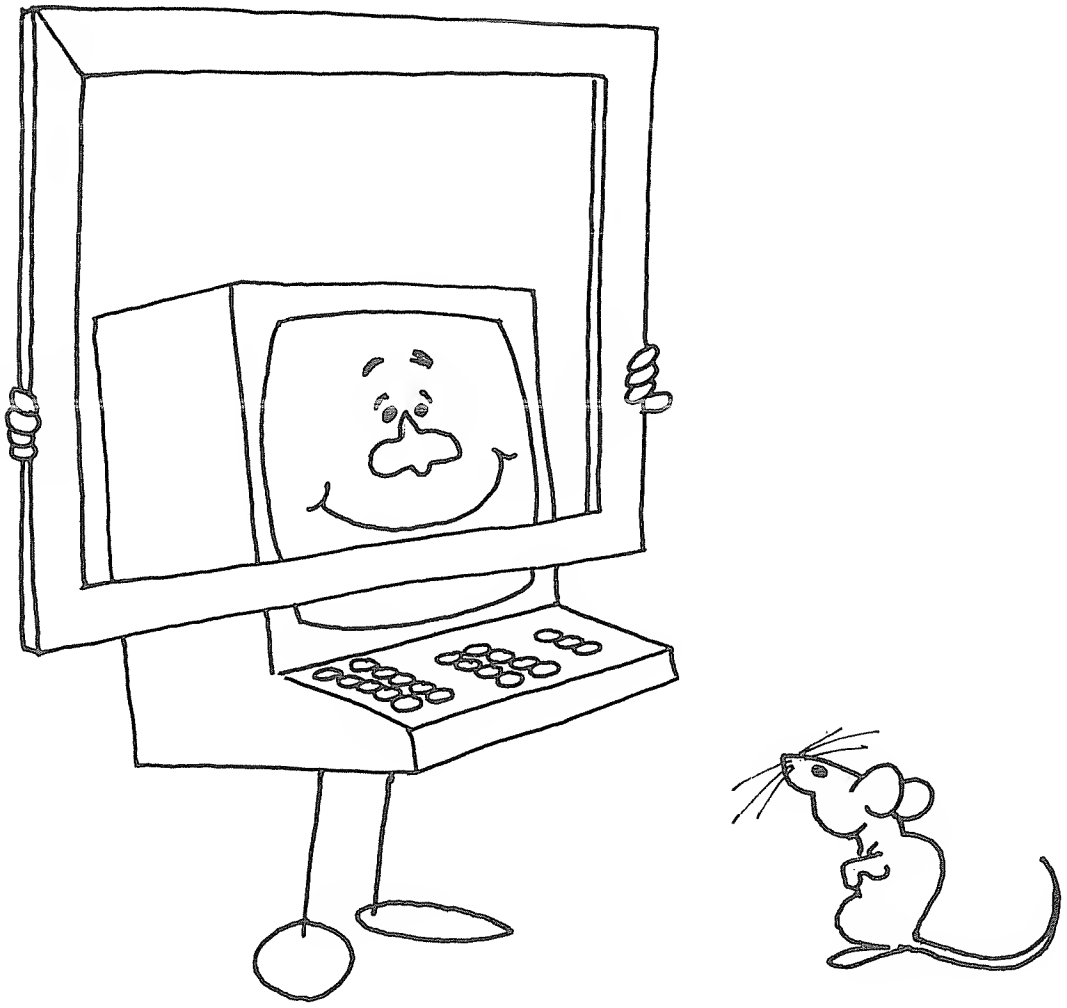
Try:

```
GOTO 20
```

This time try:

```
GOSUB 20
```

Now, can you figure out how to write your name in the frame? Try it.



24

Graphic Shapes

The TRS-80® prints a neat set of ready-made graphic shapes, from a single block to a larger block with four or six small blocks. To see the whole range of graphic shapes and their numbers, type in this program:

```
 2 CLS
10 FOR I = 128 TO 191
20 PRINT USING ''###=! '';I,CHR$(I)
30 NEXT
40 GOTO 40
```

Try the program with three spaces between ! and '' in line 20. Now try it with four spaces. What happens?

The numbers from 192 to 255 give you blanks:

192 is zero blanks

193 is one blank

194 is two blanks

195 is three blanks

. . . . . .

255 is 63 blanks.

To see this, change line 10 to:

```
FOR I = 192 TO 255
```

GLOSSARY

BREAK Stops the program from running and tells you the line number at which you stopped the program. The cursor will appear at the bottom left of the screen.

CATALOG Shows a list of all the names of programs on a disk.

CHARACTER Any one letter, number (0-9), space, or symbol.

CLS Clears everything off the screen and puts the cursor at the top left of the screen. It does not erase the memory.

COMMAND Tells the computer specific things to do. Examples include PRINT, LIST, FLASH, and INVERSE.

CONT Stands for "continue." If you stop a program with BREAK, you can continue from where you left off, if you don't change the program between BREAK and CONT.

CURSOR The white blinking rectangle that moves on the screen. It moves one line below whatever you are writing.

DATA A list of values to be given to the READ statements.

DELETE Erases a program from a disk by typing DELETE and the program name.

FOR/NEXT The FOR statement tells the computer to do something a certain number of times. The NEXT statment comes after the FOR statement and tells the computer to go back and do the action stated. The FOR/NEXT statement is a type of loop.

Example:
```
10 FOR A = 1 to 15
15 PRINT A
20 NEXT A
```

GOSUB The command to tell the computer to go to a subroutine, which can be used over and over again in the program.

GOTO Tells the program to go to the line number stated and the program continues from that line.

Example:
```
GOTO 100
```

IF-THEN If the relationship of two numbers or two strings is true, then the computer will do whatever is commanded. If false, it will go to the next line.

Example:
```
110 IF A=B THEN PRINT ''A DOES EQUAL B''
120 IF R$= ''HI'' THEN GOTO 500
```

INITIALIZE (INIT) Initializes a disk using the following steps:

1. Insert the System Master Disk or any initialized disk into the disk drive and turn on the power.
2. After the red light on the disk drive goes off, remove the disk and insert a blank disk into the disk drive.
3. Type "INIT HELLO".
4. Remove the initialized disk and place a sticker at the top of the disk so you know it is initialized.

INPUT Enables the program user to type in a response while the program is running.

Example:
```
10 INPUT A
RUN
? 5
```

INT (or INTEGER) A command that tells the computer to round off to a whole number, or integer.

LINE NUMBER A number typed at the beginning of a memory line.

LIST Shows everything in the computer's memory in line-number order.

LOAD Loads a program from a disk into the computer's memory by typing LOAD and the program's name.

LOOP A series of instructions that are repeated over and over again until an ending condition is reached.

MEMORY Where information is stored in locations inside the computer.

MENU A list of numbered choices in a program. It asks the program user to select one.

NEW Erases the memory.

OUTPUT What you see on the screen, printer, or other hardware.

PRINT (?) A command that tells the computer to write on the screen whatever follows "PRINT".

Example:
```
? ''HELLO''
? A
? A$
```

PROGRAM Instructions telling the computer what to do and how to do it.

RANDOM (RND(0)) A command that allows the computer to choose from a range of expressions between 0 and 1.

READ Puts values from DATA statements to variable(s).

Example:
```
100 READ A, B$
```

RESET Stops the program from running and puts the cursor at the bottom left of the screen. The program stays in the memory.

RETURN Pushing the RETURN key takes whatever has been typed into the computer to be processed accordingly.

RND See Random

RUN A command that starts the program's execution.

SAVE Puts a program onto a disk by typing the command SAVE and the program's name.

SET Used in low resolution graphics to place a small square at any designated point on the screen.

SQR Computer language for "square root."

STRING One or more characters in quotation marks.

TIMER Delays a program. It leaves a certain amount of time between one part of a program and another point.

VARIABLE A character that can take on any of a given set of values.

INDEX

Kids Working with Computers

THE **ACORN® BASIC** MANUAL

THE **APPLE® BASIC** MANUAL

THE **APPLE® LOGO®** MANUAL

THE **ATARI® BASIC** MANUAL

THE **COMMODORE® BASIC** MANUAL

THE **COMMODORE® LOGO** MANUAL

THE **IBM® BASIC** MANUAL

THE **IBM® LOGO** MANUAL

THE **TEXAS INSTRUMENTS BASIC** MANUAL

THE **TEXAS INSTRUMENTS LOGO** MANUAL

THE **TRS-80® BASIC** MANUAL

THE **TRS-80® COLOR LOGO** MANUAL

CHILDRENS PRESS
REINFORCED BINDING

ISBN 0-516-08431-3